



BAB II

TINJAUAN PUSTAKA

2.1 Teori Umum

2.1.1 Pengertian Perangkat Lunak

Ladjamudin (2013:20), mengemukakan *software* merupakan kumpulan dari perintah/fungsi yang ditulis dengan aturan tertentu untuk memerintahkan komputer melaksanakan tugas tertentu. Hal senada juga dikemukakan oleh Husda dan Wangdra (2016:15), *software* adalah program yang berisi perintah-perintah untuk melakukan pengolahan data. Sedangkan S. Rosa. A dan Shalahuddin (2016:2), menyatakan perangkat lunak (*software*) adalah program komputer yang terasosiasi dengan dokumentasi perangkat lunak seperti dokumentasi kebutuhan, model desain, dan cara penggunaan (*user manual*).

Dari beberapa definisi di atas penulis menyimpulkan bahwa perangkat lunak adalah suatu program komputer yang tersusun atas perintah-perintah untuk menjalankan suatu tugas tertentu sesuai dengan perintah yang diberikan oleh pengguna komputer.

2.1.2 Pengertian Komputer

Puspitosari (2013:1), mengemukakan komputer dapat didefinisikan sebagai sekumpulan alat elektronik yang saling terkoordinasi satu sama lain sehingga dapat menerima data, kemudian mengolah data, dan pada akhirnya akan menghasilkan suatu keluaran yang berupa informasi (*Input-Proses-Output*). Hal senada juga diungkapkan oleh Husda dan Wangdra (2016:2), bahwa komputer merupakan alat untuk mengolah data sesuai perintah yang sudah dirumuskan. Sedangkan Kadir (2013:2), menyatakan komputer adalah peralatan elektronik yang biasa dipakai orang untuk membantu pelaksanaan pekerjaan.

Dari beberapa definisi di atas penulis menyimpulkan bahwa komputer adalah alat elektronik yang dapat menerima dan mengelola data sehingga menghasilkan informasi.



2.1.3 Pengertian Data

Indrajani (2015:69), mengemukakan data merupakan fakta mentah tentang orang, tempat, kejadian, dan apapun yang penting bagi perusahaan, dimana data itu sendiri tidak memiliki arti. Selain itu, Husda dan Wangdra (2016:13), menyatakan data merupakan sesuatu yang belum mempunyai arti bagi penerimanya dan masih memerlukan adanya suatu pengolahan. Sedangkan Ladjamudin (2013:9), mengemukakan bahwa data adalah kenyataan yang menggambarkan kejadian-kejadian dan kesatuan nyata.

Dari beberapa definisi di atas penulis menyimpulkan bahwa data adalah kumpulan dari beberapa fakta tentang orang, kejadian, maupun tempat yang memerlukan suatu pengolahan agar menghasilkan informasi yang berguna.

2.2 Teori Judul

2.2.1 Pengertian Aplikasi

Sujatmiko (2012:23), menyatakan *Application* merupakan program komputer yang dibuat oleh suatu perusahaan komputer untuk membantu manusia dalam mengerjakan tugas-tugas tertentu, misalnya *Ms-Word*, *Ms-Excel*. Senada dengan pendapat Asropudin (2013:6), yaitu aplikasi (*application*) adalah software yang dibuat oleh suatu perusahaan komputer untuk mengerjakan tugas-tugas tertentu, misalnya *Ms-Word*, *Ms-Excel*. Sedangkan Budiharto (2013:5), mengemukakan *Applications* merupakan program yang dapat berjalan di komputer tersendiri (*stand alone computer*), dari mulai program yang simple sampai dengan program besar dan rumit.

Berdasarkan pendapat di atas, maka dapat disimpulkan bahwa pengertian aplikasi adalah program komputer yang berfungsi untuk membantu pengguna komputer dalam mengerjakan dan menyelesaikan tugas-tugas tertentu.

2.2.2 Pengertian Perhitungan

Saptika (2015:511), perhitungan adalah proses, cara, perbuatan menghitung. Selain itu menurut Kamus Besar Bahasa Indonesia (2009:328) perhitungan adalah perbuatan, pertimbangan dsb.



2.2.3 Pengertian Sisa Hasil Usaha (SHU)

Ningsih (2007:41), Sisa Hasil Usaha adalah sisa keuntungan yang diperoleh koperasi setelah dikurangi penyusutan dan biaya-biaya yang dikeluarkan dalam mengelola koperasi.

2.2.4 Pengertian Koperasi Konsumen Warmart Veteran Utama

Koperasi Konsumen Warmart Veteran Utama Palembang adalah Koperasi yang bergerak dibidang retail yang didirikan pada tanggal 10 November 2017.

2.2.5 Pengertian Aplikasi Perhitungan Sisa Hasil Usaha (SHU) pada Koperasi Konsumen Warmart Veteran Utama

Aplikasi Perhitungan Sisa Hasil Usaha (SHU) pada Koperasi Konsumen Warmart Veteran Utama adalah suatu aplikasi yang digunakan untuk melakukan perhitungan sisa hasil usaha investor pada Koperasi Konsumen Warmart Veteran Utama yang menghasilkan output berupa hasil perhitungan sisa hasil usaha yang akan diinformasikan pada investor.

2.3 Teori Khusus



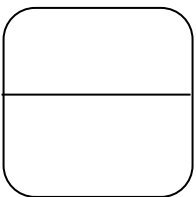
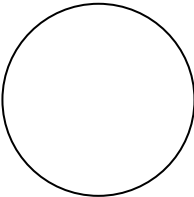
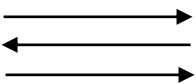
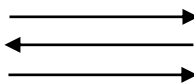
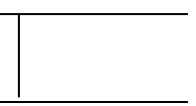
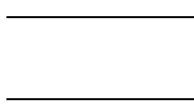
2.3.1 Pengertian DFD (*Data Flow Diagram*)

Indrajani (2015:27), mengemukakan Data Flow Diagram (DFD) adalah sebuah alat yang menggambarkan aliran data sampai sebuah sistem selesai, dan kinerja atau proses perusahaan dilakukan dalam sistem tersebut. Hal senada juga dikemukakan Husda dan Wangdra (2016:109), Data Flow Diagram (DFD) atau Diagram Alir Data (DAD) atau adalah suatu diagram yang menggunakan notasi-notasi untuk menggambarkan arus dari data sistem, yang penggunaannya sangat membantu untuk memahami sistem secara logika, terstruktur dan jelas. Selain itu, Saputra (2013:118), menjelaskan Data Flow Diagram atau yang disingkat DFD merupakan suatu diagram yang menggambarkan alir data dalam suatu entitas ke sistem atau sistem ke entitas. DFD juga dapat diartikan sebagai teknik grafis yang menggambarkan alir data dan transformasi yang digunakan sebagai perjalanan data dari input atau masukan menuju keluaran atau output.



DFD itu mempunyai 4 simbol dalam masing-masing versi, diantaranya menurut Gane/Sarson serta Yourdon/De Marco. Berikut ini daftarnya.

Tabel 2.1 Simbol-Simbol *Data Flow Diagram (DFD)*

Gane/Sarson	Yourdon/De Marco	Nama Simbol	Keterangan
		Entitas Eksternal	Entitas eksternal dapat berupa orang/unit terkait yang berinteraksi dengan sistem, tetapi di luar sistem.
		Proses	Orang, unit yang mempergunakan atau melakukan transformasi data. Komponen fisik tidak diidentifikasi.
		Aliran Data	Aliran data dengan arah khusus dari sumber ke tujuan.
		Data Store	Penyimpanan data atau tempat data di-refer oleh proses.

Sumber : Saputra (2013:118-119)

S. Rosa. A dan Shalahuddin (2016:72-73), menjelaskan berikut ini adalah tahapan-tahapan perancangan dengan menggunakan DFD:

1. Membuat DFD Level 0 atau sering disebut juga Context Diagram

DFD Level 0 menggambarkan sistem yang akan dibuat sebagai suatu entitas tunggal yang berinteraksi dengan orang maupun sistem lain. DFD Level 0



digunakan untuk menggambarkan interaksi antara sistem yang akan dikembangkan dengan entitas luar.

2. Membuat DFD Level 1

DFD Level 1 digunakan untuk menggambarkan modul-modul yang ada dalam sistem yang akan dikembangkan. DFD Level 1 merupakan hasil *breakdown* DFD Level 0 yang sebelumnya sudah dibuat.

3. Membuat DFD Level 2

Modul-modul pada DFD Level 1 dapat di-*breakdown* menjadi DFD Level 2. Modul mana saja yang harus di-*breakdown* lebih detail tergantung pada kedetailan modul tersebut. Apabila modul tersebut sudah cukup detail dan rinci maka modul tersebut sudah tidak perlu di-*breakdown* lagi. Untuk sebuah sistem, jumlah DFD Level 2 sama dengan jumlah modul pada DFD Level 1 yang di-*breakdown*.

4. Membuat DFD Level 3 dan seterusnya


DFD Level 3,4,5, dan seterusnya merupakan *breakdown* dari modul pada DFD Level di-atasnya. *Breakdown* pada level 3,4,5, dan seterusnya aturannya sama persis dengan DFD Level 1 atau level 2.

2.3.2 Blockchart

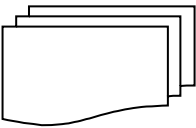
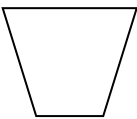

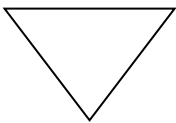
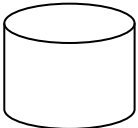

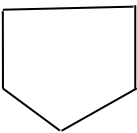
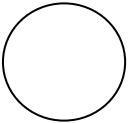

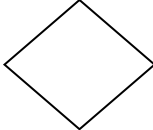
Kristanto (2008:75) *BlockChart* berfungsi untuk memodelkan masukan, keluaran, proses maupun transaksi dengan menggunakan simbol-simbol tertentu.

Adapun simbol-simbol yang sering digunakan dalam *blockchart* dapat dilihat pada tabel berikut ini:


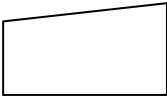
Tabel 2.2. Simbol-Simbol dalam *Blockchart*

No	Simbol	Keterangan
1.		Menandakan dokumen, bisa dalam bentuk surat, formulir, buku/ bendel/ berkas atau cetakan.

Lanjutan Tabel 2.2. Simbol-Simbol dalam *Blockchart*

No	Simbol	Keterangan
2.		Multi dokumen
3.		Proses Manual
4.		Proses yang dilakukan oleh komputer.
5.		Menandakan dokumen yang diarsipkan (arsip manual)
6.		Data penyimpanan (<i>Storage</i>)
7.		Proses apa saja yang tidak terdefinisi termasuk aktivitas fisik.
8.		Terminasi yang mewakili simbol tertentu untuk digunakan pada aliran lain pada halaman yang lain.
9.		Terminal yang mewakili simbol tertentu untuk digunakan pada aliran lain pada halaman yang sama.
10.		Terminal yang menandakan awal dan akhir dari suatu aliran.
11.		Pengambilan keputusan (<i>Decision</i>).

Lanjutan Tabel 2.2. Simbol-Simbol dalam *Blockchart*

No	Simbol	Keterangan
12.		Layar peraga (<i>monitor</i>).
13.		Pemasukkan data secara manual.

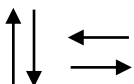


Sumber : Kristanto (2008:75-77)

2.3.3 Bagan Alir (*Flowchart*)

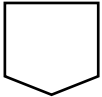

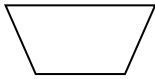
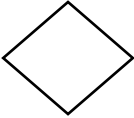
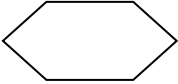


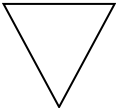



Indrajani (2015:36), menyatakan *Flowchart* merupakan penggambaran secara grafik dari langkah-langkah dan urutan prosedur suatu program. Sama halnya menurut Saputra (2013:120), *Flowchart* merupakan suatu diagram yang menggambarkan alur kerja suatu sistem. Selain itu Ladjamudin (2013:263), mengemukakan *Flowchart* adalah bagan-bagan yang mempunyai arus yang menggambarkan langkah-langkah penyelesaian suatu masalah.

Ladjamudin (2013:266-268), *flowchart* disusun dengan simbol. Simbol ini dipakai sebagai alat bantu menggambarkan proses di dalam program. Simbol-simbol yang digunakan yakni sebagai berikut:

Tabel 2.3. Simbol-Simbol dalam *Flowchart*

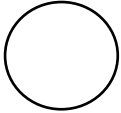



No	Simbol	Keterangan
1.		Simbol arus/flow Untuk menyatakan jalannya arus suatu proses
2.		Simbol Communication link Untuk menyatakan bahwa adanya transisi suatu data/informasi dari satu lokasi ke lokasi lainnya
3.		Simbol Connector Untuk menyatakan sambungan dari satu proses ke proses lainnya dalam halaman/lembar yang sama.

Lanjutan Tabel 2.3. Simbol-Simbol dalam *Flowchart*

No	Simbol	Keterangan
4.		Simbol Offline Connector Untuk menyatakan sambungan dari proses ke proses lainnya dalam halaman yang berbeda.
5.		Simbol Offline Connector Untuk menyatakan sambungan dari proses ke proses lainnya dalam halaman yang berbeda.
6.		Simbol Manual Untuk menyatakan suatu tindakan (proses) yang tidak dilakukan oleh komputer (manual).
7.		Simbol Decision/logika Untuk menunjukkan suatu kondisi tertentu yang akan menghasilkan dua kemungkinan jawaban, ya / tidak
8.		Simbol Predefined Proses Untuk menyatakan penyediaan tempat penyimpanan suatu pengolahan untuk memberi harga awal.
9.		Simbol Terminal Untuk menyatakan permulaan atau akhir suatu program
10.		Simbol Keying Operating Untuk menyatakan segala jenis operasi yang diproses dengan menggunakan suatu mesin yang mempunyai keyboard.
11.		Simbol off-line storage Untuk menunjukkan bahwa data dalam symbol ini akan disimpan ke suatu media tertentu
12.		Simbol Manual input Untuk memasukkan data secara manual dengan menggunakan online keyboard.
13.		Simbol Input-output Untuk menyatakan proses input dan output tanpa tergantung dengan jenis peralatannya.
14.		Simbol Punched Card Untuk menyatakan input berasal dari kartu atau output ditulis ke kartu.



Lanjutan Tabel 2.3. Simbol-Simbol dalam *Flowchart*

No	Simbol	Keterangan
15.		Simbol Magnetic-tape unit Untuk menyatakan bahwa adanya transisi suatu data/informasi dari satu lokasi ke lokasi lainnya
16.		Simbol Disk storage Untuk menyatakan input berasal dari disk atau output disimpan ke disk.
17.		Simbol Document Untuk mencetak laporan ke printer.
18.		Simbol Display Untuk menyatakan peralatan output yang digunakan berupa layar (video, komputer).

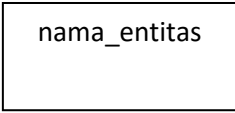
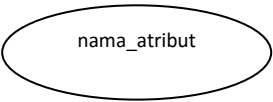
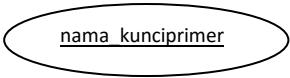
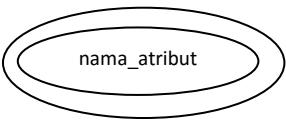
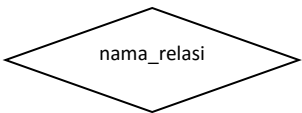
Sumber : Ladjamudin (2013:266-268)

2.3.4 Pengertian ERD (Entity Relational Diagram)

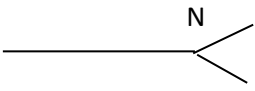
Ladjamudin (2013:142), ERD adalah suatu model jaringan yang menggunakan susunan data yang disimpan dalam sistem secara abstrak”. Sedangkan Pratama (2014:49), mengemukakan ERD (*Entity Relationship Diagram*) adalah diagram yang menggambarkan keterkaitan antar tabel beserta dengan *field-field* di dalamnya pada suatu database sistem. Selain itu menurut S. Rosa. A dan Shalahuddin (2016:50), menjelaskan pemodelan awal basis data yang paling banyak digunakan adalah menggunakan *Entity Relationship Diagram* (ERD). ERD dikembangkan berdasarkan teori himpunan dalam bidang matematika. ERD digunakan untuk pemodelan basis data relasional. ERD memiliki beberapa aliran notasi seperti notasi Chen (dikembangkan oleh Peter Chen), Barker (dikembangkan oleh Richard Barker, Ian Palmer, Harry Ellis), notasi Crow’s Foot, dan beberapa notasi lain. Namun yang banyak digunakan adalah notasi dari Chen. Berikut adalah simbol-simbol yang digunakan pada ERD dengan notasi Chen:



Tabel 2.4. Simbol-simbol pada ERD

No	Simbol	Keterangan
1.	Entitas/ <i>entity</i> 	Entitas merupakan data inti yang akan disimpan; bakal tabel pada basis data; benda yang memiliki data dan harus disimpan datanya agar dapat diakses oleh aplikasi komputer; penamaan entitas biasanya lebih ke kata benda dan belum merupakan nama tabel.
2.	Atribut 	<i>Field</i> atau kolom data yang butuh disimpan dalam suatu entitas
3.	Atribut kunci primer 	<i>Field</i> atau kolom data yang butuh disimpan dalam suatu entitas dan digunakan sebagai kunci akses <i>record</i> yang diinginkan; biasanya berupa id; kunci primer dapat lebih dari satu kolom, asalkan kombinasi dari beberapa kolom tersebut dapat bersifat unik (berbeda tanpa ada yang sama)
4.	Atribut multivalai / multivalue 	<i>Field</i> atau kolom data butuh disimpan dalam satu entitas yang dapat dimiliki nilai lebih dari satu.
5.	Relasi 	Relasi yang menghubungkan antar entitas; biasanya diawali dengan kata kerja.

**Lanjutan Tabel 2.4.** Simbol-Simbol *Entity Relationship Diagram* (ERD)

No	Simbol	Keterangan
6.	Asosiasi / <i>association</i> 	Penghubung antara relasi dan entitas di mana di kedua ujungnya memiliki <i>multiplicity</i> kemungkinan jumlah pemakaian. Kemungkinan jumlah maksimum keterhubungan antara entitas satu dengan entitas yang lain disebut dengan kardinalitas. Misalkan ada kardinalitas 1 ke N atau sering disebut dengan <i>one to many</i> menghubungkan entitas A dan entitas B

Sumber : S. Rosa. A dan Shalahuddin (2016:50-51)

2.3.5 Pengertian Kamus Data

S. Rosa. A dan Shalahuddin (2016:73), mengemukakan kamus data adalah kumpulan daftar elemen data yang mengalir pada sistem perangkat lunak sehingga masukan (*input*) dan keluaran (*output*) dapat dipahami secara umum (memiliki standar cara penulisan). Hal senada juga dikemukakan oleh Kristanto (2008:72), kamus data adalah kumpulan elemen-elemen atau simbol-simbol yang digunakan untuk membantu dalam penggambaran atau pengidentifikasian setiap field atau file di dalam sistem. Selain itu Indrajani (2015:30), menngemukakan kamus data adalah katalog fakta tentang data dan kebutuhan informasi suatu sistem informasi.

S. Rosa. A dan Shalahuddin (2016:74), menjelaskan kamus data memiliki beberapa simbol untuk menjelaskan informasi tambahan sebagai berikut:

Tabel 2.5. Simbol-simbol dalam Kamus Data

No	Simbol	Arti
1	=	disusun atau terdiri atas
2	+	dan
3	[]	baik ...atau...

**Lanjutan Tabel 2.5.** Simbol-simbol dalam Kamus Data

No	Simbol	Arti
4	{ } ⁿ	n kali diulang/ bernilai banyak
5	()	data operasional
6	*...*	batas komentar

Sumber : S. Rosa. A dan Shalahuddin (2016:74)

2.3.6 Metode Waterfall

S. Rosa. A dan Shalahuddin (2016:28), metode SDLC air terjun (*waterfall*) sering juga disebut model sekuensial linier (*sequential linear*) atau alur hidup klasik (*classic life cycle*). Model air terjun menyediakan pendekatan alur hidup perangkat lunak secara sekuensial atau terurut mulai dari analisis, desain, pengodean, pengujian, dan tahap pendukung (*support*).

a. Analisis kebutuhan perangkat lunak

Tahap analisis dilakukan secara intensif untuk menspesifikasikan kebutuhan sistem agar dapat dipahami sistem seperti apa yang dibutuhkan oleh *user*.

b. Desain

Tahap desain adalah proses multi langkah yang fokus pada desain pembuatan program sistem termasuk struktur data, arsitektur sistem, representasi antarmuka, dan prosedur pengodean. Tahap ini mentranslasi kebutuhan sistem dari tahap analisis kebutuhan ke representasi desain agar dapat diimplementasikan menjadi program pada tahap selanjutnya.

c. Pembuatan kode program

Pada tahap pengkodean, desain harus ditranslasikan ke dalam program sistem. Hasil dari tahap ini adalah program komputer sesuai dengan desain yang telah dibuat pada tahap desain.

d. Pengujian

Tahap pengujian fokus pada sistem dari segi logika dan fungsional dan memastikan bahwa semua bagian sudah diuji. Hal ini dilakukan untuk meminimalisir kesalahan (*error*) dan memastikan keluaran yang dihasilkan sesuai dengan yang diinginkan.



e. Pendukung (*support*) atau Pemeliharaan (*maintenance*)

Tidak menutup kemungkinan sebuah sistem mengalami perubahan ketika sudah dikirimkan ke *user*. Perubahan bisa terjadi karena adanya kesalahan yang muncul dan tidak terdeteksi saat pengujian atau sistem harus beradaptasi dengan lingkungan baru. Tahap pemeliharaan dapat mengulangi proses pengembangan mulai dari analisis spesifikasi untuk perubahan sistem yang sudah ada, tapi tidak untuk sistem baru.

2.3.7 Pengujian *Black Box Testing*

S. Rosa. A dan Shalahuddin (2016:275), *black box testing* (pengujian kotak hitam) yaitu menguji perangkat lunak dari segi spesifikasi fungsional tanpa menguji desain dan program. Pengujian dimaksudkan untuk mengetahui apakah fungsi-fungsi, masukan, dan keluaran dari perangkat lunak sesuai dengan spesifikasi yang dibutuhkan.

2.4 Teori Program

2.4.1 Basis Data (*Database*)

Indrajani (2015:70), basis data adalah sebuah kumpulan data yang saling berhubungan secara logis, dan merupakan sebuah penjelasan dari data tersebut, yang didesain untuk menemukan data yang dibutuhkan oleh sebuah organisasi. Selain itu, Everest (dikutip Husda dan Wangdra 2016:119), juga mengemukakan basis data adalah koleksi atau kumpulan data yang mekanis terbagi/shared, terdefinisi secara formal dan dikontrol terpusat pada organisasi. Hal yang senada juga dikemukakan oleh Badiyanto (2013:57), *database* diartikan sebagai suatu kumpulan dari data yang tersimpan dalam tabel dan diatur atau diorganisasikan sehingga data tersebut bisa diambil atau dicari dengan mudah dan efisien.

Dari beberapa definisi di atas penulis menyimpulkan bahwa basis data (*database*) adalah tempat yang digunakan untuk menyimpan data-data agar lebih terorganisir antara satu sama lain sehingga mempermudah dalam proses pencarian data.



2.4.1.1 Pengertian *MySQL*

MySQL merupakan tipe data relasional yang artinya *MySQL* menyimpan datanya dalam bentuk tabel-tabel yang saling berhubungan, Winarno, dkk (2014:102). Hal senada juga dikemukakan Sadeli (2014:10), *MySQL* adalah database yang menghubungkan *script php* menggunakan perintah *query* dan *escaps character* yang sama dengan *php*. Selain itu Saputra (2013:14), juga mengemukakan *MySQL* merupakan database storage engine yang paling banyak digunakan oleh web developer karena sifatnya yang free, alias gratis.

2.4.1.2 Keuntungan *MySQL*

Winarno, dkk (2014:102), menjelaskan ada banyak database untuk *PHP*, namun *MySQL* merupakan software database yang paling disarankan. Berikut ini adalah keuntungan *MySQL* :

1. Gratis dan *open source*.
2. Ada versi komersialnya juga, digunakan jika ingin memberikan dukungan teknis.
3. Biaya yang harus dikeluarkan jauh lebih murah dibandingkan merek lainnya.
4. Tersedia di banyak platform.
5. Menggunakan standar penulisan *SQL ANSI*.

2.4.2 Web Editor

Sadeli (2014:12), mengemukakan bahwa *web editor* adalah program yang dapat mempermudah pekerjaan programmer dalam mendesain tampilan aplikasi berbasis web ataupun membuat kode program.

Penulis menggunakan *Adobe Dreamweaver CS3* sebagai web editor dalam membangun Aplikasi Perhitungan SHU sebagai tempat pengkodean yang akan menghubungkan database agar terhubung dengan program yang akan dibuat. Penulis menggunakan *Adobe Dreamweaver CS3* karena *Adobe Dreamweaver CS3* merupakan web editor yang memiliki banyak keunggulan dan juga fitur-fitur yang terdapat didalamnya mudah digunakan oleh penulis.



2.4.2.1 Adobe Dreamweaver

Dreamweaver merupakan suatu perangkat lunak web editor keluaran Adobe System yang digunakan untuk membangun dan mendesain suatu website dengan fitur-fitur yang menarik dan kemudahan dalam penggunaannya, Sadeli (2014:12-13). Selain itu Madcoms (2016:14), menjelaskan *dreamweaver* adalah software aplikasi desain web visual yang biasa dikenal dengan istilah WYSIWYG (*What You See Is What You Get*), intinya tidak harus berurusan dengan tag-tag HTML untuk membuat sebuah situs. Hal yang senada juga dikemukakan oleh Madcoms (2012:2), *dreamweaver* adalah sebuah editor profesional yang menggunakan HTML untuk mendesain web secara visual dan mengelola situs atau halaman web.

2.4.3 Bahasa Pemrograman

Abdulloh (2016:1), bahasa pemrograman merupakan bahasa yang dapat dipahami oleh komputer. Selain itu, Husda dan Wangdra (2016:42) juga mengemukakan bahasa pemrograman adalah software bahasa komputer yang digunakan dengan merancang atau membuat program sesuai dengan struktur dan metode yang dimiliki oleh program itu sendiri.

2.4.3.1 HTML (*HyperText Markup Language*)

2.4.3.1.1 Pengertian HTML

Abdulloh (2016:2), menjelaskan HTML singkatan dari *HyperText Markup Language*, yaitu skrip berupa tag-tag untuk membuat dan mengatur struktur website. Selain itu, Winarno, dkk (2014:1), juga menjelaskan *HyperText Markup Language* (HTML) adalah sebuah bahasa menampilkan konten di Web. Hal senada juga dikemukakan (Asropudin, 2013:45), bahwa *HyperText Markup Language* adalah bahasa komputer yang digunakan untuk membuat sebuah halaman *webpage*.



2.4.3.1.2 Struktur Dasar HTML

Budiharto (2013:27), menjelaskan dokumen HTML merupakan dokumen web yang statis, artinya hanya mampu menampilkan teks dan gambar yang statis sehingga untuk membuatnya lebih dinamis (misal: teks, dan gambar bergerak, menampilkan database suatu perusahaan, dan lainnya) harus menggunakan bahasa lainnya seperti Javascript, Java, PHP, dll. Command atau perintah dalam HTML disebut juga dengan tag. Suatu halaman HTML mempunyai format seperti berikut ini:

Welcome.html:

```
<!DOCTYPE html>
<html>
<head>
<title> Halaman Web Pertamaku </title>
</head>
<body>
<h1>Saya mahasiswa BINUS University </h1><br>
<p> We really proud to learn Web Programming with Mr. Widodo.</p>
</body>
</html>
```

2.4.3.2 PHP (*Hypertext Preprocessor*)

2.4.3.2.1 Pengertian PHP (*Hypertext Preprocessor*)

PHP: *Hypertext Preprocessor* adalah bahasa skrip yang dapat ditanamkan atau disisipkan ke dalam HTML/PHP banyak dipakai untuk membuat situs web dinamis, (Badiyanto, 2013:32). Selain itu, Winarno, dkk (2014:63), juga menjelaskan PHP atau PHP *Hypertext Preprocessor* adalah sebuah bahasa pemrograman web berbasis server (server-side) yang mampu parsing kode PHP dari kode web dengan ekstensi .php, sehingga menghasilkan tampilan website yang dinamis di sisi client. Hal senada juga dikemukakan Raharjo (2016:38), bahwa PHP adalah salah satu bahasa pemrograman skrip yang dirancang untuk membangun aplikasi web.



2.4.3.2.2 Tipe Data dalam PHP (*Hypertext Preprocessor*)

Winarno dkk (2014:65), menjelaskan bahwa tipe data PHP berbeda dengan bahasa pemrograman lain, variable di PHP lebih fleksibel, anda tidak perlu mendefinisikan jenisnya ketika mendefinisikan pertama kali.

Ada 6 variabel dasar yang dapat diakomodasikan di PHP, seperti terlihat ditabel:

Tabel 2.6 Tipe Data pada PHP

Tipe	Contoh	Penjelasan
Integer	134	Semua angka bukan pecahan
Double	5.123	Nilai pecahan
String	“asep”	Kumpulan karakter
Boolean	False	Salah satu nilai true atau false
Object		Sebuah instansi dari class
Array		Larik

Sumber : Winarno dkk (2014:65)

2.4.3.2.3 Skrip PHP

Badiyanto (2013:32-33), menjelaskan bahwa *PHP* yang merupakan sebuah bahasa scripting yang terpasang pada *HTML*, dan skrip *HTML* (*Hyper Text Markup Language*) adalah bahasa standar untuk membuat halaman *web*. Biasa file dituliskan dengan ekstensi *.htm* atau *.html*.

Contoh:

```
<HTML>
<HEAD>
<TITLE> Latihan HTML </TITLE>
</HEAD>
<BODY>
    Mari belajar membuat web
</BODY>
</HTML>
```



Contoh di atas bisa ditulis dengan menggunakan *PHP* sebagai berikut yang kodenya di simpan dengan latihan1.php.

```
<HTML>
<HEAD>
<TITLE> Latihan HTML </TITLE>
</HEAD>
<BODY>
  <?php
    Printf ("Mari Belajar Membuat Web");
    // atau
    Echo "<br>";
    Echo "Mari Belajar Membuat Web";
  ?>
</BODY>
</HTML>
```

2.4.4 Webservice

Untuk bisa mengakses aplikasi web yang akan dibuat oleh penulis, maka diperlukan yang namanya *webservice*. Sadeli (2014:2), mengemukakan bahwa *web server* adalah sebuah perangkat lunak server yang berfungsi menerima permintaan HTTP dan HTTPS dari klien yang dikenal dengan web browser dan mengirimkan kembali hasilnya dalam bentuk halaman-halaman web yang umumnya berbentuk dokumen *HTML*. Selain itu Pratama (2014:439), mengemukakan bahwa *web server* merupakan perangkat lunak yang dijalankan di sistem operasi pada komputer server maupun desktop yang berfungsi menerima permintaan (*request*) dalam protokol". Dan hal senada juga dikemukakan oleh Madcoms (2016:4), *web server* adalah suatu program computer yang mempunyai tanggung jawab atau tugas menerima permintaan HTTP dari komputer klien, yang dikenal dengan nama *web browser* dan melayani mereka dengan menyediakan respon HTTP berupa konten data, biasanya berupa halaman web yang terdiri dari dokumen



HTML dan objek terkait seperti gambar dan lain-lainnya. Penulis menggunakan *Xampp* sebagai *webserver* dalam membangun Aplikasi Perhitungan SHU.

2.4.4.1 Pengertian XAMPP

Xampp adalah paket program web lengkap yang dapat anda pakai untuk belajar pemrograman web, khususnya *PHP* dan *MySQL*, paket ini dapat didownload secara gratis dan legal Nugroho (2013:1). Hal yang senada juga dikemukakan oleh Sidik (2014:72), bahwa *XAMPP* (*X(Windows/Linux) Apache MySQL PHP dan Perl*) merupakan paket *server web PHP* dan *database MySQL* yang paling populer di kalangan pengembang web dengan menggunakan *PHP* dan *MySQL* sebagai databasenya. Selain itu Riyanto (2014:11), berpendapat bahwa, *XAMPP* merupakan paket *PHP* dan *MYSQL* berbasis *open source* yang dapat digunakan digunakan sebagai tool pembantu pengembangan aplikasi berbasis *PHP*.

Berdasarkan pendapat di atas, dapat disimpulkan bahwa *XAMPP* merupakan paket *server web PHP* dan *database MySQL* yang terdiri dari *Apache*, *MySQL*, *PHP* dan *phpMyAdmin*

2.4.4.2 Folder Utama Xampp

Nugroho (2013:7), menjelaskan bahwa dibawah folder utama *Xampp*, terdapat beberapa folder penting yang perlu diketahui. Untuk lebih memahami setiap fungsinya, Anda dapat melihat penjelasannya sebagai berikut:

1. *Apache*
Folder utama dari *Apache Webserver*.
 2. *Htdocs*
Folder utama untuk menyimpan data-data latihan web, baik *PHP* maupun *HTML* biasa. Pada folder ini, anda dapat membuat subfolder sendiri untuk mengelompokkan file latihannya. Semua folder dan file program di *htdocs* bisa diakses dengan mengetikkan alamat <http://localhost/> di browser.
 3. *Manual*
-



Berisi subfolder yang di dalam terdapat manual program dan database, termasuk manual *PHP* dan *MYSQL*.

4. *Mysql*

Folder utama untuk *database MYSQL server*. Di dalamnya terdapat subfolder data (lengkapnya: C:\xampp\mysql\data) untuk merekam semua nama database, serta subfolder bin yang berisi tools klien dan server *MYSQL*.

5. *Php*

Folder utama untuk program *PHP*.